UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/697,820	10/29/2003	Robert Cochran	100203007-1	9527

22879 7590 11/21/2006

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER	
LU, KUEN S	

ART UNIT	PAPER NUMBER
2167	

DATE MAILED: 11/21/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/697,820	COCHRAN ET AL.	
	Examiner	Art Unit	
	Kuen S. Lu	2167	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 August 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-32* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-32* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☑ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date ____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendments

1. This Action is responsive to Applicant's Amendment, filed on August 29, 2006. As to Applicant's Arguments/Remarks filed August 29, 2006, please see discussion in "***Response to Arguments***", following this Action for Final Rejection (hereafter "the Action"), shown next. Please note the Action maintains the same grounds of rejection as set forth in the Office Action for non-Final Rejection (hereafter "the non-Final"), mailed January 7, 2006, and, therefore **claims 1 and 2-32 remain pending**. In view of amendments made to claims 1, 9, 17, 22 and 28, rejections made to the claims in the non-Final under 35 USC § 101 and 35 U.S.C. §112, 1st paragraph is hereby withdrawn.

Drawings

2. The drawings filed October 29, 2003 have been accepted.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

3.1. Claims 1-32 are rejected under 35 U.S.C. 103(a) as being unpatentable over Baird: Oracle 8i Data Guard Concepts, Administration, and Installation Guide, Release 3.0, October 2001, Oracle® (hereafter "OraDgd") in view of Bobrowski et al.: Oracle7™ Server Concepts, Release 7.3, February 1996, Oracle® (hereafter "Ora734").

As per claim 1, OraDgd teaches "A database system capable of executing a database application that transfers a logical object in multiple fragments, the database system" (See Fig. 1-1 and Pages 1-8 and 1-9 wherein archived redo logs of a production database are shipped to standby database site and applied database changes to the standby database) comprising:

"a main storage site" (See Page 1-16 disk space on both production and standby database sites are monitored); and

"a remote storage site adapted to link to the main storage site and to receive and store mirror information stored in the main storage site, the remote storage site including a storage and a cache sidefile" (See Page 1-35 OraDgd teaches a remote mirror callout feature allowing mirroring of online redo logs and at Pages 1-33 and 1-34 where a cache file is implemented at the standby database site to store information for restarting and rolling back failover and switchover).

OraDgd does not explicitly teach that the remote cache sidefile is "divided into a plurality of array sidefile recordsets".

However, Ora734, at Pages 22-17 and 24-15, teaches log writer writing commit record immediately into redo log buffer where atomic write of a database transaction

record is assigned with an entry number, the system change number and each redo log file includes a plurality of transaction records.

It would have been obvious to one having ordinary skill in the art at the time of the Applicant's invention was made to combine the teaching of Ora734 with the OraDgd reference by utilizing cache file for storing and mirroring online redo logs because both references are directed to database implementation and the combined teaching would have efficiently achieved a no-data-loss failover since online redo logs are mirrored at the remote database site for being available anytime to be applied to the standby database in order to synchronize standby database with production database (See OraDgd: Page 1-24).

The combined teaching of the Ora734 and OraDgd references further teaches the following:

"a main protocol executable on the main storage site and adapted to transfer the logical object in multiple fragments in combination with information indicative of logical object fragment commencement and completion in the multiple fragment database application transfer" (See OraDgd: Page 1-9 shipper at the production host site ships archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery); and

"a remote protocol executable on the remote storage site and adapted to control the cache sidefile to cache the multiple fragments as received and to destage the logical object to the storage on receipt of all fragments" (See OraDgd: Page 1-9 Applier at the remote standby database site applies changes from each archived redo logs to the standby database wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery).

As per claim 9, OraDgd teaches "An article of manufacture" (See Fig. 1-1 and Page 1-8 where Oracle8i Data Guard Architecture is an article of manufacture" comprising: "a controller usable medium having a computable readable program code embodied therein for executing in a database system that runs a database application for mirroring a logical object in multiple fragments from a main storage site to a remote storage site, the computable readable program code" (See Pages 1-8, 1-9 and 1-35 where database is replicated to remote standby database site and OraDgd further teaches a remote mirror callout feature allowing mirroring of online redo logs) further comprising: "a code configured to cause the controller to interface with the database application that links and mirrors data between the main storage site and the remote storage site, the remote storage site including a storage and a cache sidefile" (See Page 1-35 OraDgd teaches a remote mirror callout feature allowing mirroring of online redo logs and at

Pages 1-33 and 1-34 where a cache file is implemented at the standby database site to store information for restarting and rolling back failover and switchover).

OraDgd does not explicitly teach that the remote cache sidefile is "divided into a plurality of array sidefile recordsets".

However, Ora734, at Pages 22-17 and 24-15, teaches log writer writing commit record immediately into redo log buffer where atomic write of a database transaction record is assigned with an entry number, the system change number and each redo log file includes a plurality of transaction records.

It would have been obvious to one having ordinary skill in the art at the time of the Applicant's invention was made to combine the teaching of Ora734 with the OraDgd reference by utilizing cache file for storing and mirroring online redo logs because both references are directed to database implementation and the combined teaching would have efficiently achieved a no-data-loss failover since online redo logs are mirrored at the remote database site for being available anytime to be applied to the standby database in order to synchronize standby database with production database (See OraDgd: Page 1-24).

The combined teaching of the Ora734 and OraDgd references further teaches the following:

"a code configured to cause the controller to create and deploy transfer the logical object in multiple fragments in combination with control information indicative of logical object fragment commencement and completion in the multiple fragment database application transfer" (See OraDgd: Page 1-9 shipper at the production host site ships

archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery);

"the control information controlling the cache sidefile to cache the multiple fragments as received and to destage the logical object to the storage on receipt of all fragments" (See OraDgd: Page 1-9 Applier at the remote standby database site applies changes from each archived redo logs to the standby database wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery).

As per claim 17, OraDgd teaches "An article of manufacture" (See Fig. 1-1 and Pages 1-8 and 1-9 wherein archived redo logs of a production database are shipped to standby database site and applied database changes to the standby database) comprising:

a controller usable medium having a computable readable program code embodied therein for executing in a database system that runs a database application for mirroring a logical object in multiple fragments from a main storage site to a remote storage site, the computable readable program code" (See Pages 1-8, 1-9 and 1-35 where database

is replicated to remote standby database site and OraDgd further teaches a remote mirror callout feature allowing mirroring of online redo logs) further comprising: "a code executable at the remote storage site configured to cause the controller to control storage of the logical object multiple fragments in a cache sidefile" (See Page 1-35 OraDgd teaches a remote mirror callout feature allowing mirroring of online redo logs and at Pages 1-33 and 1-34 where a cache file is implemented at the standby database site to store information for restarting and rolling back failover and switchover).

OraDgd does not explicitly teach that the remote cache sidefile is "divided into a plurality of array sidefile recordsets".

However, Ora734, at Pages 22-17 and 24-15, teaches log writer writing commit record immediately into redo log buffer where atomic write of a database transaction record is assigned with an entry number, the system change number and each redo log file includes a plurality of transaction records.

It would have been obvious to one having ordinary skill in the art at the time of the Applicant's invention was made to combine the teaching of Ora734 with the OraDgd reference by utilizing cache file for storing and mirroring online redo logs because both references are directed to database implementation and the combined teaching would have efficiently achieved a no-data-loss failover since online redo logs are mirrored at the remote database site for being available anytime to be applied to the standby database in order to synchronize standby database with production database (See OraDgd: Page 1-24).

The combined teaching of the Ora734 and OraDgd references further teaches the following:

"a code executable at the remote storage site configured to cause the controller to receive the logical object in multiple fragment transfers in combination with control information indicative of logical object fragment commencement and completion" (See OraDgd: Page 1-9 shipper at the production host site ships archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery); and

"a code executable at the remote storage site configured to cause the controller to cache the multiple fragments as received and to destage the logical object to the storage on receipt of all fragments" (See OraDgd: Page 1-9 Applier at the remote standby database site applies changes from each archived redo logs to the standby database wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery).

As per claim 22, OraDgd teaches "A storage element readable by a controller tangibly embodying a program of instructions executable by the controller to perform method acts for executing in a database system that runs a database application for mirroring a logical object in multiple fragments from a main storage site to a remote storage site" (See Pages 1-8, 1-9 and 1-35 where database is replicated to remote standby database site and OraDgd further teaches a remote mirror callout feature allowing mirroring of online redo logs), the method acts comprising:
"controlling storage of the logical object multiple fragments at the remote storage site in a cache sidefile" (See Page 1-35 OraDgd teaches a remote mirror callout feature allowing mirroring of online redo logs and at Pages 1-33 and 1-34 where a cache file is implemented at the standby database site to store information for restarting and rolling back failover and switchover).

OraDgd does not explicitly teach that the remote cache sidefile is "divided into a plurality of array sidefile recordsets".

However, Ora734, at Pages 22-17 and 24-15, teaches log writer writing commit record immediately into redo log buffer where atomic write of a database transaction record is assigned with an entry number, the system change number and each redo log file includes a plurality of transaction records.

It would have been obvious to one having ordinary skill in the art at the time of the Applicant's invention was made to combine the teaching of Ora734 with the OraDgd reference by utilizing cache file for storing and mirroring online redo logs because both references are directed to database implementation and the combined teaching would

have efficiently achieved a no-data-loss failover since online redo logs are mirrored at the remote database site for being available anytime to be applied to the standby database in order to synchronize standby database with production database (See OraDgd: Page 1-24).

The combined teaching of the Ora734 and OraDgd references further teaches the following:

"receiving the logical object at the remote storage site in multiple fragment transfers in combination with control information indicative of logical object fragment commencement and completion" (See OraDgd: Page 1-9 shipper at the production host site ships archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery); and

"caching the multiple fragments at the remote storage site as received and to enable destaging of the logical object at the remote storage site to the storage on receipt of all fragments" (See OraDgd: Page 1-9 Applier at the remote standby database site applies changes from each archived redo logs to the standby database wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery).

As per claim 28, OraDgd teaches "A storage element readable by a controller tangibly embodying a program of instructions executable by the controller to perform method acts for executing in a database system that runs a database application for mirroring a logical object in multiple fragments from a main storage site to a remote storage site" (See Pages 1-8, 1-9 and 1-35 where database is replicated to remote standby database site and OraDgd further teaches a remote mirror callout feature allowing mirroring of online redo logs), the method acts comprising:
"interfacing with the database application that links and mirrors data between the main storage site and the remote storage site, the remote storage site including a storage and a cache sidefile(See Page 1-35 OraDgd teaches a remote mirror callout feature allowing mirroring of online redo logs and at Pages 1-33 and 1-34 where a cache file is implemented at the standby database site to store information for restarting and rolling back failover and switchover).

OraDgd does not explicitly teach that the remote cache sidefile is "divided into a plurality of array sidefile recordsets".

However, Ora734, at Pages 22-17 and 24-15, teaches log writer writing commit record immediately into redo log buffer where atomic write of a database transaction record is assigned with an entry number, the system change number and each redo log file includes a plurality of transaction records.

It would have been obvious to one having ordinary skill in the art at the time of the Applicant's invention was made to combine the teaching of Ora734 with the OraDgd

reference by utilizing cache file for storing and mirroring online redo logs because both references are directed to database implementation and the combined teaching would have efficiently achieved a no-data-loss failover since online redo logs are mirrored at the remote database site for being available anytime to be applied to the standby database in order to synchronize standby database with production database (See OraDgd: Page 1-24).

The combined teaching of the Ora734 and OraDgd references further teaches the following:

"deploying from the main storage site the logical object in multiple fragments in combination with control information indicative of logical object fragment commencement and completion in the multiple fragment database application transfer" (See OraDgd: Page 1-9 shipper at the production host site ships archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery);

"the control information controlling the cache sidefile to cache the multiple fragments as received and to destage at the remote storage site the logical object to the storage on receipt of all fragments" (See OraDgd: Page 1-9 Applier at the remote standby database site applies changes from each archived redo logs to the standby database wherein the time archive log file created is indicated to and part of file attributes, and further at last

paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery).

As per claims 2 and 10, the combined teaching of the Ora734 and OraDgd references further teaches "the main protocol includes information indicative of logical object fragment commencement and completion using a technique selected from among a group" (See OraDgd: Page 1-9 shipper at the production host site ships archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery) consisting of:
"(1) explicitly sending a start control message preceding the multiple fragments and an end control message concluding the multiple segments" (See Ora734: Pages 12-5, 22-6 and 24-15 where log writer writes system change numbers to on-line redo log which may be explicitly used to reconstruct all changes made to the database), and
(2) implicitly determining either a start control message and an end control message" (See Ora734: Page 22-7 where log writer writes switches to next online redo log file when current file is filled which implicitly indicating a minimum and a maximum system change numbers in each file).

As per claim 18, 23 and 29, Ora734 further teaches "creating control information indicative of logical object fragment commencement and completion using a technique selected from among a group" (See Ora734: Page 22-7 where system change numbers along with transaction entries are created in the online redo log) consisting of: "(1) receiving explicitly identified starting and ending fragments" (See Ora734: Pages 12-5, 22-6 and 24-15 where log writer writes system change numbers to on-line redo log which may be explicitly used to reconstruct all changes made to the database), and "(2) deriving either of the starting fragment and the ending fragment implicitly from received control information" (See Ora734: Page 22-7 where log writer writes switches to next online redo log file when current file is filled which implicitly indicating a minimum and a maximum system change numbers in each file).

As per claims 3 and 11, Ora734 further teaches "an address translation process that translates a logical address to a list of physical addresses" (See Ora734: Page 6-9 where rowid of a database record translate the logical address of a record into physical address).

As per claims 4, 12 and 24, Ora734 further teaches "an address translation process that resolves a virtual write address of the database application into a pick list of actual physical media writes associated with the logical object" (See Ora734: Page 6-9 where rowid is assigned to database record).

As per claims 5, 13 and 25, Ora734 further teaches "a process adapted to create a control message for communication to the remote protocol that instructs individual physical storage elements to operate on the multiple physical writes as a single object entity so that all or none is destaged to the storage" (See OraDgd: Page 1-20, 1-21, 4-18 and 4-19 where messages are sent to production and standby databases when normal mode commands are processed, applier invokes shipper to ship archived logs, applier check archived logs and applied the logs to the standby database).

As per claims 6, 14 and 26, the combined teaching of the Ora734 and OraDgd references further teaches the following:
"a process adapted to receive an application request to write the logical object of a specified length to a specified virtualized storage address" (See OraDgd: Page 1-28 where all archived redo logs and online redo logs are applied to the database, and Ora734: last paragraphs of Pages 22-17 and 24-15 system change numbers of redo log entries and time may be specified for the length of database recovery);
"a process adapted to convert the virtualized write address and resolving the transfer length to designate at least one physical address in at least one physical storage device for transferring the logical object in fragments" (See OraDgd: Pages 1-16 and 2-7 where archived log files are compressed and shipped to standby database site, the standby database site uncompressed the file and applies the file to standby database);
"a process adapted to send a first control message to the at least one physical storage device that delineates the start of a logical object that is to be held in a remote mirror

cache for destaging" (See OraDgd: Page 1-9 shipper at the production host site ships archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery); and

"a process adapted to send a second control message that delineates the end of the logical object so that the mirror cache is destaged to the at least one physical storage device, no portion of the logical object fragments being otherwise destaged" (See OraDgd: Page 1-9 shipper at the production host site ships archived redo logs to the standby database site wherein the time archive log file created is indicated to and part of file attributes, and further at last paragraphs of Pages 22-17 and 24-15 of Ora734 reference teaches that archived log is a copy of online redo logs preserving system change number of each entry, along with time data of the entry, serving as bases for database recovery).

As per claims 7, 15, 20, 27 and 31, Ora734 further teaches "information is replicated from the main storage site to the remote storage site using a technique selected from among a group including: (1) synchronous data replication and (2) asynchronous data replication" (See Pages 15-3 and 24-5 where synchronous data replication and asynchronous I/O operations are supported).

As per claims 8, 16, 21 and 32, the combined teaching of the Ora734 and OraDgd references further teaches "the logical object multiple fragments are controllably destaged in all-or-none fashion to all devices in a consistency group" (See OraDgd: Pages 1-28 and 2-34 where shipper and applier work consistently and all archived redo logs and online redo logs are applied to the database, and Ora734: last paragraphs of Pages 22-17 and 24-15 system change numbers of redo log entries and time may be specified for the length of database recovery).

As per claims 19 and 30, the combined teaching of the Ora734 and OraDgd references further teaches "a code adapted to cause the controller to track order of fragment updating between the main storage site and the remote storage site including updating of the sidefile recordsets" (See OraDgd: Page 1-20, 1-21, 4-18 and 4-19 where messages are sent to production and standby databases when normal mode commands are processed, applier invokes shipper to ship archived logs, applier check archived logs and applied the logs to the standby database, and Ora734: Page 22-7 where system change numbers along with transaction entries are created in the online redo log).

8. The prior art made of record

U. Baird: Oracle 8i Data Guard Concepts, Administration, and Installation Guide, Release 3.0, October 2001, Oracle®

V. Bobrowski et al.: Oracle7™ Server Concepts, Release 7.3, February 1996, Oracle®

The prior art made of record and not relied upon is considered pertinent to Applicant's disclosure.

A. U.S. Patent Application 2004/0230859

B. U.S. Patent 6,735,636

C. U.S. Patent 6,636,908

Response to Arguments

4. The Applicant's arguments filed on August 29, 2006 has been fully considered, please see discussion below:

a). At pages 11-12, concerning claims 1-8, Applicant argued that Oracle Data Guard (hereafter "Data Guard") reference does not teach "the remote storage site including a storage and a cache sidefile divided into a plurality of array sidefile recordsets".

As to the above argument **a)**, Examiner respectfully submits that the combined teaching of Oracle Data Guard's internal transaction cache file serving as transaction cache and Oracle7 Server Concepts' (hereafter "Concepts") log writer writing commit record immediately into redo log buffer where atomic write of a database transaction record assigned with an entry number, the system change number and each redo log file includes a plurality of transaction records provides an equivalent teaching. Also please note the transaction cache is located at remote site, the standby database. (See Page Glossary-4, "transaction cache".

b). At page 11, concerning claims 1-8, Applicant continued to argue that the atomic write in Data Guard and Concepts references is about forming a redo log and consolidate information into queue at the primary or destination. However, application describes multiple fragments of a logical objects are sent from main storage site to the remote site along with information identifying starting and ending of the fragments.

As to the above argument **b)**, Examiner respectfully submits that Oracle references does provide an equivalent teaching as described follow: an SCN represents an atomic change (See Concepts: Page 12-15) for transaction to commit. Log writer writes on-line redo log with the SCN entries. Data change for a committed transaction is not necessarily immediately written to the log file because of consideration of efficiency (See Concepts: Page 12-15). It is within the teaching of Data Guard and Concepts that change of each transaction be written to an on-line redo log file and archived log file for shipping to standby site without consolidating data change of some number of transactions. The Data Guard and Concepts references teach **an equivalent of** multiple fragments of a logical objects are sent from main storage site to the remote site along with information identifying starting and ending of the fragments **by** identifying all SCN entries in a log file being shipped from primary to standby site.

c). At page 12, concerning claims 9-16, Applicant made a similar argument as in item **b)**.

As to the above item **c)**, Examiner respectfully applies the same ground that SCN

entries, starting with a lowest and ending with a highest, identifies data changes of transactions in each archived log file shipped from a primary site to a standby site. Furthermore, each log file may consist of data change related to a sole transaction.

d). At page 13, concerning claims 17-32, Applicant further argued about deficiency of Data Guard and Concepts on teaching caching multiple fragments as received and to destage the logical object to the storage on receipt of all fragments.

As to the above argument **d),** Examiner respectfully suggests the same response as previously described in response to item **a)**.

Conclusions

5. Applicant's amendment necessitated the new grounds of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1 .136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Contact information

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kuen S Lu whose telephone number is (571) 272-4114. The examiner can normally be reached on Monday-Friday (8:00 am-5:00 pm). If attempts to reach the examiner by telephone pre unsuccessful, the examiner's Supervisor, John Cottingham can be reached on (571) 272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for Page 13 published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 886-217-9197 (toll-free).

Kuen S. Lu K.S. Lu

Patent Examiner, Art Unit 2167

November 13, 2006

JOHN COTTINGHAM
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100